

Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/JP05/001519

International filing date: 02 February 2005 (02.02.2005)

Document type: Certified copy of priority document

Document details: Country/Office: JP
Number: 2004-025474
Filing date: 02 February 2004 (02.02.2004)

Date of receipt at the International Bureau: 29 September 2005 (29.09.2005)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)



World Intellectual Property Organization (WIPO) - Geneva, Switzerland
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse

日 本 国 特 許 庁
JAPAN PATENT OFFICE

07. 2. 2005

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日
Date of Application: 2 0 0 4 年 2 月 2 日

出 願 番 号
Application Number: 特 願 2 0 0 4 - 0 2 5 4 7 4

パリ条約による外国への出願
に用いる優先権の主張の基礎
となる出願の国コードと出願
番号

The country code and number
of your priority application,
to be used for filing abroad
under the Paris Convention, is

J P 2 0 0 4 - 0 2 5 4 7 4

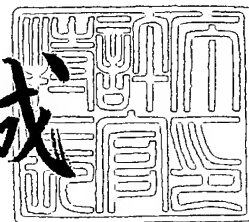
出 願 人
Applicant(s): 大 下 誠

2 0 0 5 年 9 月 1 5 日

特許庁長官
Commissioner,
Japan Patent Office

中 嶋

誠



【書類名】 特許願
【あて先】 特許庁長官殿
【国際特許分類】 G06F 17/21
【発明者】
 【住所又は居所】 東京都世田谷区若林 4 丁目 3 0 番地 1 0 号 岡元ビル 2 B
 【氏名】 大下 誠
【特許出願人】
 【識別番号】 303035891
 【住所又は居所】 東京都世田谷区若林 4 丁目 3 0 番地 1 0 号 岡元ビル 2 B
 【氏名又は名称】 大下 誠
【手数料の表示】
 【予納台帳番号】 222624
 【納付金額】 21,000円
【提出物件の目録】
 【物件名】 特許請求の範囲 1
 【物件名】 明細書 1
 【物件名】 図面 1
 【物件名】 要約書 1

【書類名】 特許請求の範囲

【請求項 1】

変更箇所を開始部・中間部・終了部の三つに抽象化することで、テンプレートをプレゼンテーションデータとプレゼンテーションロジックの二つに分離したテンプレートシステム

【書類名】明細書

【発明の名称】 プレゼンテーションデータとプレゼンテーションロジックとを分離したテンプレートシステム

【技術分野】**【0001】**

本発明は、変更箇所を開始部・中間部・終了部の三つに抽象化することで、プレゼンテーション層を表すテンプレートを「プレゼンテーションデータ」と「プレゼンテーションロジック」とに分離し、両者の独立性を高めたテンプレートシステムである。

【背景技術】**【0002】**

何らかの表示や出力を伴うソフトウェアプログラムは、内部を次の二つに分けて考えることができる。

- ・ビジネスロジック層…「何を」表示するかを司る部分
- ・プレゼンテーション層…「どのように」表示するかを司る部分

【0003】

例えば「従業員の一覧を表示するプログラム」を考えると、次のようになる。

- ・「何を」表示するか（名前、社員番号、内線番号など）を決め、そのデータを用意するのがビジネスロジック層
- ・用意されたデータを「どのように」表示するか（表示形式、文字の大きさ、色など）を決め、表示するのがプレゼンテーション層

【0004】

ビジネスロジック層とプレゼンテーション層を分離するための手段として「テンプレートシステム」がある。テンプレートシステムとは、表示する形式のデータをテンプレート（ひな型）として用意し、これをメインプログラムから読み込み、必要な部分を書き換えて出力する仕組みである。

【0005】

つまり、テンプレートシステムでは次のように役割分担を行うことで、ビジネスロジック層とプレゼンテーション層とを分離している。

- ・メインプログラムがビジネスロジック層（何を表示するか）を担当する。
- ・テンプレートがプレゼンテーション層（どうやって表示するか）を担当する。

【0006】

テンプレートシステムを用いてビジネスロジック層とプレゼンテーション層を分離すると、次のような利点がある。

- ・プレゼンテーション層（テンプレート）を切替えるだけで、様々な表示形式に対応することができる。このとき、ビジネスロジック層（メインプログラム）は変更する必要がない。
- ・一方の変更が他方に影響を与えず互いに独立して変更できるため、メインプログラムを担当するプログラマーとテンプレートを担当するデザイナーとが分業しやすくなる。

【0007】

このような特徴から、テンプレートシステムは特に、表示画面の変更が頻繁に発生するWebアプリケーションにおいて利用されている。

【非特許文献1】 Velocity、<http://jakarta.apache.org/velocity/>

【非特許文献2】 XMLC、<http://xmlc.enhydra.org/>

【発明の開示】**【発明が解決しようとする課題】****【0008】**

実は、プレゼンテーション層（つまりテンプレート）はさらに次の二つに分けて考えることができる。

- ・プレゼンテーションデータ…表示に必要なデータ
- ・プレゼンテーションロジック…表示に必要なロジック（繰り返し、条件分岐など）

【0009】

例えば、従業員の一覧を表形式で表示することを考える。このとき、

- ・表の大きさ、枠の幅や色などがプレゼンテーションデータ
- ・従業員の数だけ表示を「繰り返す」というのがプレゼンテーションロジック

と考えることができる。

(注意して欲しいのだが、プレゼンテーションロジックはあくまで「表示のために必要なロジック」である。同じロジックでも、データを用意するのが目的であるビジネスロジックとは役割が異なる。そのため、プレゼンテーションロジックをビジネスロジック層に含めることはできない。)

【0010】

従来のテンプレートシステムでは、ビジネスロジック層(メインプログラム)とプレゼンテーション層(テンプレート)とを分離することを目的としていた。そのため、プレゼンテーション層であるテンプレートの中に、プレゼンテーションデータとプレゼンテーションロジックが混在しており、両者が分離されていなかった(図1の1)。

【0011】

両者が分離できていないと、次のような不都合が発生する。

- ・一方の変更を行ったときに、誤ってもう一方も変更してしまう可能性が高い。
- ・プレゼンテーションデータだけ、あるいはプレゼンテーションロジックだけを取り出して再利用することができない。
- ・デザイナーは一般にロジックの記述が苦手である。そのためプレゼンテーションロジックが複雑になったとき、テンプレートを担当しているデザイナーでは対応できない。

【課題を解決するための手段】

【0012】

本発明では、テンプレートを「プレゼンテーションデータ」と「プレゼンテーションロジック」の二つに分離することにより、上記の問題を解決する。

【0013】

そのために、次のような手段をとる。

- ・テンプレートを二つに分けて、「プレゼンテーションデータ」(図1の2)と「プレゼンテーションロジック」(図1の3)を別々に記述する。
- ・プレゼンテーションデータにおいて、変更すべき箇所には何らかの「目印」をつける(図1の2では「id="users"」が目印である)。目印がついた変更箇所は、開始部・中間部・終了部の三つに分けられる。
- ・プレゼンテーションロジックでは、目印がつけられた変更すべき箇所に対して、繰り返しや分岐などの操作を行う。操作は、開始部・中間部・終了部のそれぞれまたは全部に対して行うことができる。

【0014】

つまり、プレゼンテーションデータにおいて変更すべき箇所を開始部・中間部・終了部の三つに抽象化することで、プレゼンテーションロジックでは、例えば中間部だけを繰り返したり、三つとも繰り返したりなど、自由に操作することができるわけである。

【0015】

また目印のつけ方には以下のような方法が考えられる。

- ・プレゼンテーションデータ中に、目印となる記号やキーワードを埋め込む(図1の2における「id="users"」はこの方法である)。
- ・プレゼンテーション中に現れる文字列やパターンを目印とみなす。
- ・プレゼンテーションデータの先頭または末尾からの文字数やバイト数をもって目印とみなす。

【0016】

このようにすれば、プレゼンテーションデータの中にプレゼンテーションロジックが入ることはなく、またプレゼンテーションロジックの中にプレゼンテーションデータが入ることもない。つまり、両者の分離が可能となる。

【0017】

またこの方法はプレゼンテーションデータの形式によらず適用できる。つまり本発明はどのような形式の出力や表示にも対応できる。

【発明の効果】

【0018】

本発明により、次のような効果が得られる。

- ・プレゼンテーションデータとプレゼンテーションロジックとを完全に分離することができる。
- ・一方を変更するときに他方に影響を与えない。つまり互いが独立して変更できる上、他方を誤って変更してしまうことがない。
- ・どちらか一方だけを取り出して再利用することができる。
- ・どのような表示形式や出力系式にも対応することができる。

【発明を実施するための最良の形態】

【0019】

以下では具体例を用いて本発明の実施形態を説明する。具体例としては、HTMLファイルを出力するWebアプリケーションを考える。この場合、プレゼンテーションデータはHTML形式になる。

【0020】

(1) プレゼンテーションデータを作成し、変更する箇所に目印をつける(図2)。この例では2種類の目印が使われている。

・「id="users"」4は、それを含むHTMLタグ「」から「」までを変更箇所とみなすことを表す。

・「\$user」5は、変数userの値で置き換えることを表す。

【0021】

(2) プレゼンテーションデータから、目印をもとに4つのマクロを自動生成する(図3)。この例では次のようなマクロが自動生成される。

- ・マクロ「head_users」6…開始部()を表す。
- ・マクロ「body_users」7…中間部、つまり開始部と終了部の間の部分(#user)を表す。
- ・マクロ「foot_users」8…終了部()を表す。
- ・マクロ「elem_users」9…変更箇所全体、つまり開始部+中間部+終了部を表す。

このとき、(1)のプレゼンテーションデータをマクロelem_usersで置き換える(図3の10)。また目印である「id="users"」は取り除かれる。

【0022】

(3) プレゼンテーションロジックをマクロとして作成する(図4)。このマクロは、変更箇所全体を表すマクロ(ここではelem_users)を上書きするように作成する。この例では、配列userlistの要素をひとつひとつ変数userに代入しながら、中間部であるマクロbody_usersを繰り返している(この繰り返しがプレゼンテーションロジックである)。

【0023】

(4) 上の(2)と(3)のマクロ定義を読み込んで合成し、展開する(図5)。

(4-1) プレゼンテーションデータはelem_usersというマクロに置き換えられているので、これを展開する。

(4-2) 展開した中にプレゼンテーションロジックとマクロが含まれるので、マクロをさらに展開する。

(4-3) プレゼンテーションデータとプレゼンテーションロジックとが合成されたテンプレートが得られる。

【0024】

(5) このテンプレートをメインプログラムから読み込み、実行する。例えば配列userlistの内容が["foo", "bar", "baz"]であった場合、図6のようなHTMLファイルが生

成される。

【0025】

従来の技術では、(4)で得られるようなテンプレートを直接生成しなければならなかった。つまりプレゼンテーションデータとプレゼンテーションロジックが分離されておらず、1つになっていた。本発明では、テンプレートをプレゼンテーションデータ(図2)とプレゼンテーションロジック(図4)とに分離できていることがわかる。

【実施例1】

【0026】

図6は、開始部と終了部を含めた全体を繰り返すようにしたプレゼンテーションロジックの例である。

【0027】

図7は、開始部と終了部を出力しないようにした(つまりタグを出力しないようにした)プレゼンテーションロジックの例である。タグを出力しないようにすることで、HTMLやXML以外の形式を出力することができる。

【0028】

図8は、条件分岐など複雑なプレゼンテーションロジックを記述した例である。これらのようにプレゼンテーションロジックを複雑に変更しても、プレゼンテーションデータのほうは一切変更する必要がない。つまり、両者の分離が実現されていることがわかる。

【産業上の利用可能性】

【0029】

本発明は任意の形式のデータを出力することができるため適用範囲も広く、HTMLやXMLを出力するアプリケーションのほかに、各種設定ファイルや通信用データの生成など多岐に渡る。

【図面の簡単な説明】

【0030】

【図1】従来の方法および本発明におけるテンプレートの例

【図2】目印をつけたプレゼンテーションデータの例

【図3】プレゼンテーションデータから自動生成されたマクロの例

【図4】マクロを上書きして作成したプレゼンテーションロジックの例

【図5】マクロを合成して自動生成されたテンプレートの例

【図6】出力結果の例

【図7】プレゼンテーションロジックの例：タグを含めて繰り返す

【図8】プレゼンテーションロジックの例：タグを出力しない

【図9】プレゼンテーションロジックの例：条件分岐を含む

【符号の説明】

【0031】

- 1 … 従来の方法におけるテンプレート
- 2 … 本発明におけるテンプレート (プレゼンテーションデータ部分)
- 3 … 本発明におけるテンプレート (プレゼンテーションロジック部分)
- 4 … プレゼンテーションデータにおいて変更箇所を表す目印
- 5 … プレゼンテーションデータにおいて変数の参照を表す目印
- 6 … 変更箇所の開始部を表すマクロ
- 7 … 変更箇所の中間部を表すマクロ
- 8 … 変更箇所の終了部を表すマクロ
- 9 … 変更箇所全体を表すマクロ
- 10 … プレゼンテーションデータを変換したマクロ

【書類名】 図面

【図 1】

(A)

```
<ul>
#foreach(user in list)
  <li>${user}</li>
#end
</ul>
```

1

従来のテンプレート
(プレゼンテーションデータ+プレゼンテーションロジック)

(B)

```
<ul id="users">
  <li>${user}</li>
</ul>
```

2

プレゼンテーションデータ

```
(開始部)
#foreach(user in list)
  (中間部)
#end
(終了部)
```

3

プレゼンテーションロジック

【図 2】

```
<ul id="users">
  <li>${user}</li>
</ul>
```

4

5

【図 3】

```
#defmacro(head_users)
  <ul>
#end

#defmacro(body_users)
  <li>${user}</li>
#end

#defmacro(foot_users)
  </ul>
#end

#defmacro(elem_users)
  #expand(head_users)
  #expand(body_users)
  #expand(foot_users)
#end

#expand(elem_users)
```

6

7

8

9

10

【図 4】

```
#defmacro(elem_users)
  #expand(head_users)
  #foreach(user in userlist)
    #expand(body_users)
  #end
  #expand(foot_users)
#end
```

【図 5】

```
#expand(elem_users)
```

↓マクロを展開

```
#expand(head_users)
#foreach(user in userlist)
  #expand(body_users)
#end
#expand(foot_uesrs)
```

↓マクロを展開

```
<ul>
#foreach(user in userlist)
  <li>${user}</li>
#end
</ul>
```

【図 6】

```
<ul>
  <li>foo</li>
  <li>bar</li>
  <li>baz</li>
</ul>
```

【図 7】

```
#defmacro(elem_users)
  #foreach(user in userlist)
    #expand(head_users)
    #expand(body_users)
    #expand(foot_users)
  #end
#end
```

【図 8】

```
#defmacro(elem_users)
  #foreach(user in userlist)
    #expand(body_users)
  #end
#end
```

【図 9】

```
#defmacro(elem_users)
  #set(i = 0)
  #foreach(user in userlist)
    #set(i = i+1)
    #if(i%2 == 0)
      #set(color = "red")
    #else
      #set(color = "blue")
    #end
    #expand(head_users)
    #expand(body_users)
    #expand(foot_users)
  #end
#end
```

【書類名】 要約書

【要約】

【課題】 テンプレートシステムにおいて、ひとつのテンプレート中にプレゼンテーションデータとプレゼンテーションロジックとを混在させないために、両者を分離する。

【解決手段】 プレゼンテーションデータ 2 に対して、変更すべき箇所になんらかの目印をつける。そして、変更すべき箇所を開始部・中間部・終了部の三つに抽象化し、プレゼンテーションロジック 3 からはその抽象化された三つを操作することで、両者の分離を実現する。

【選択図】 図 1

認定・付加情報

特許出願の番号	特願 2 0 0 4 - 0 2 5 4 7 4
受付番号	5 0 4 0 0 1 6 6 7 8 4
書類名	特許願
担当官	伊藤 雅美 2 1 3 2
作成日	平成 1 6 年 2 月 3 日

<認定情報・付加情報>

【提出日】 平成16年 2月 2日

特願 2 0 0 4 - 0 2 5 4 7 4

出 願 人 履 歴 情 報

識別番号

[3 0 3 0 3 5 8 9 1]

1. 変更年月日

2 0 0 3 年 6 月 2 8 日

[変更理由]

新規登録

住 所

東京都世田谷区若林 4 - 3 0 - 1 0 岡元ビル 2 B

氏 名

大下 誠